Opérateur LIRE_CHAMP

1 But

Lire un champ dans un fichier au format MED et le stocker dans un concept de type champ.

Le champ est désigné dans le fichier par son nom et éventuellement par un paramètre temporel.

Le concept produit est du type correspondant à ce qui a été demandé.

2 Syntaxe

```
champ_lu  =  LIRE_CHAMP   (

   ◊  FORMAT    =  'MED'  ,                                  [DEFAUT]

    ♦  TYPE_CHAM    =  /  'NOEU_TEMP_R' ,
                       /  'NOEU_DEPL_R' ,
                       /  'ELGA_SIEF_R' ,
                       /  etc ...

   ♦  MAILLAGE  =  ma  ,                                     [maillage]
    ♦  NOM_MED =   nommed ,                                  [TXM]

    ◊     # si TYPE-CHAM =  'ELxx_yyyy'
      ◊  MODELE=modele                                       [modele]
       ◊  PROL_ZERO=  / 'NON'                                [DEFAUT]
                      / 'OUI'


   ◊  #  Sélection du paramètre temporel
       / NUME_ORDRE = ordre  ,                               [I]
      /  NUME_PT = pas de temps ,                            [I]
      /  INST = inst,                                        [R]
          ◊  / CRITERE   =      'RELATIF'                    [DEFAUT]
                ◊  PRECISION =  /  PREC,                     [R]
                                 / 1.0E-6,                   [DEFAUT]
             / CRITERE   =      'ABSOLU'
                ♦  PRECISION =     PREC,                     [R]


   #  Choix des composantes à lire : les mêmes que le champ dans ASTER
      ou une liste imposée

   ◊  /  NOM_CMP_IDEM    =  'OUI'  ,                         [TXM]
      /  ♦  NOM_CMP      =  lcmp   ,                         [l_TXM]
         ♦  NOM_CMP_MED  =  lcmpmed   ,                      [l_TXM]

   ◊  NOM_MAIL_MED = nomamed ,                               [TXM]

   ◊  UNITE =   /  unite  ,                                  [I]
                /  81  ,                                     [DEFAUT]

   ◊  INFO   =  /  1   ,                                     [DEFAUT]
                /  2   ,
)
```

3 Opérandes

3.1 Opérande FORMAT

◊ `FORMAT = 'MED'`

Choix du format du fichier contenant le champ à lire.

Remarque :

Seul le format `MED` est opérationnel aujourd'hui. Cependant, avec l'enchaînement de `LIRE_TABLE` puis `CREA_CHAMP/EXTR_TABLE` on peut lire un champ stocké dans une table au format `ASTER`

3.2 Opérande TYPE_CHAM

```
♦ TYPE_CHAM =      /  'NOEU_TEMP_R' ,
                   /  'NOEU_DEPL_R' ,
                   /  'ELGA_SIEF_R' ,
                   /  etc ...
```

On désigne ici le type du concept à produire. Le nom du type est construit avec la logique habituelle *Code_Aster*. Les qautre premiers caractères sont `'NOEU'`, `'ELEM'`, `'ELNO'` ou `'ELGA'`. On trouve ensuite `'_'`. La séquence suivante définit le type de champ : `'TEMP'`, `'DEPL'`, `'SIEF'`, etc ... Le nom se termine par `'_R'`, `'_F'` ou `'_C'` selon le type informatique des valeurs.

Exemple : `'NOEU_TEMP_R'` , `'NOEU_DEPL_R'` , `'ELGA_SIEF_R'` etc …

Attention :

Il n'y a aucun contrôle de cohérence ! On peut très bien créer un concept température en relisant un champ qui était un déplacement à l'écriture du fichier.

Remarque :

Il peut arriver que des valeurs lues dans le fichier ne soient pas affectées dans le champ final. Par exemple, si on lit un champ de pression sur des éléments TETRA4 alors qu'il doit être affecté sur des mailles de bord (car c'est sa nature), on sera averti par ce type d'alarme :

```
<A> <LIRE_RESU> <LRCEME>
    VALEURS NON AFFECTEES DANS LE CHAMP : 3699
    VALEURS LUES DANS LE FICHIER        : 3699
```

3.3 Opérande MAILLAGE

♦ `MAILLAGE = ma`

Nom du maillage ASTER sur lequel sera exprimé le champ à lire.

3.4 Opérande NOM_MED

♦ `NOM_MED = nommed`

Nom selon la convention MED du champ à lire dans le fichier. C'est une chaîne de 32 caractères.

3.5 Opérande MODELE

◊ MODELE = mo

Nom du modèle ASTER sur lequel sera exprimé le champ à lire. Cet opérande est obligatoire si le champ à lire est un champ « par éléments » (TYPE_CHAM='ELxx_yyyy')

3.6 Opérande PROL_ZERO

◊ PROL_ZERO = 'NON' / 'OUI'

Lorsque l'on crée un champ « par éléments », la structure de ce champ est imposée par Aster. Si par exemple, on crée un champ de contraintes « ELNO » sur un modèle 3D, tous les nœuds des éléments doivent porter les composantes SIXX, SIYY, …, SIYZ. Si le champ MED que l'on lit ne possède pas toutes les valeurs attendues pas Aster, il faut « inventer » ces valeurs manquantes. La valeur « inventée » sera 0. si PROL_ZERO='OUI', elle sera « NaN » (Not a Number) si PROL_ZERO='NON'

3.7 Sélection du paramètre temporel NUME_ORDRE, NUME_PT, INST, CRITERE, PRECISION

Si le champ a été écrit dans le fichier sans référence à un paramètre temporel, rien n'est à mentionner dans cette commande de lecture. Sinon il faut préciser de quel instant il s'agit. Cela se fait par la désignation d'un numéro d'ordre, de pas de temps ou d'une valeur d'un instant d'archivage. Se référer au document [U4.71.00] pour les détails sur ces mots-clés.

3.8 Opérande NOM_CMP_IDEM

◊ / NOM_CMP_IDEM = 'OUI'

Indique qu'on doit lire dans le fichier MED les composantes dont le même nom apparaît dans la liste des composantes du champ au sens du *Code_Aster*.

3.9 Opérandes NOM_CMP, NOM_CMP_MED

```
◊ / ♦ NOM_CMP     = lcmp ,
    ♦ NOM_CMP_MED = lcmpmed ,
```

Ces deux listes doivent être de même longueur. On lit dans le fichier MED les composantes listées dans lcmpmed, puis on les affecte dans les composantes au sens de *Code_Aster*, de même rang dans la liste lcmp.

3.10 Opérande NOM_MAIL_MED

◊ NOM_MAIL_MED = nomamed

Si cet opérande est absent, on cherche le champ lié au premier maillage dans le fichier. C'est ce qui se passe quand le fichier ne contient qu'un seul maillage.

Si le fichier contient plusieurs maillages, on précise ici lequel est associé au champ que l'on veut lire.

3.11 Opérande UNITE

◊ UNITE = unite

Numéro de l'unité logique du fichier, correspond à la valeur fournie dans astk ou en utilisant la commande DEFI_FICHIER.

3.12 Opérande INFO

```
◊   INFO   =  /  1 ,        [DEFAUT]
              /  2 ,
```

Si INFO vaut 2, quelques impressions de déboggage ont lieu.
Sinon, rien n'a lieu

4 Exemple

```
temp2  =  LIRE_CHAMP   (
                MAILLAGE     =  m2,
                NOM_MED      =  'RESUUN__TEMP',
                NUME_ORDRE   =  2,
                TYPE_CHAM    =  'NOEU_TEMP_R',
                         )
```

Cette commande créera un champ aux nœuds de nom temp2 et de type NOEU_TEMP_R. Le maillage support est m2. Les valeurs sont celles stockées sous le nom RESUUN__TEMP dans le fichier MED fourni sur l'unité 81 avec le numéro d'ordre 2.